信用交渉における公開木戦略の計算量(計算機科学の理論とその応用)

AUTHOR(S):

山本, 有輝; 高田, 喜朗; 関, 浩之

CITATION:

山本, 有輝 ...[et al]. 信用交渉における公開木戦略の計算量(計算機科学の理論とその応用). 数理解析研究所講究録 2007, 1554: 276-283

ISSUE DATE:

2007-05

URL:

http://hdl.handle.net/2433/80941

数理解析研究所講究録
第1554巻 2007年 276-283

信用交渉における公開木戦略の計算量

山本 有輝† 高田 喜朗† 関 浩之†
†奈良先端科学技術大学院大学 情報科学研究科
〒630-0192 奈良県生駒市高山町8916-5
E-mail: †{yuki-ya,y-takata,seki}@is.naist.jp

あらまし 信用交渉とは，サービス提供者とサービス要求者の両者が信任状の交換を繰り返して徐々に信用を確立する，信用管理のアプローチのひとつである．信用交渉における戦略とは，それまでに交換した信任状と自身のポリシーに対して，次に相手に公開する信任状を返すような写像である．Yuらは，DTファミリーという戦略の集合を提案し，それが戦略集合として望ましい性質を満たすことを示した．DTS（公開木戦略）は，DTファミリーの中で，相手に公開する情報が最も少ない，すなわち最も慎重な戦略である．DTSは単純な実装では指数的な計算量がかかるが，効率のよい実装が存在するかどうかや，計算量の下界は，これまで知られていなかった．本研究では，Yuらの枠組の再定式化を行い，DTSの計算量の上界および下界について考察した．その結果，Yuらの定義に従った場合にはDTSはNP困難であること，また，交渉を成功に導くことに貢献しない出力を除外するよう条件を変えた場合には多項式時間可解であることがわかった．

キーワード 信用交渉，交渉戦略，デジタル信任状，計算量

1. まえがき

ネットワーク上で不特定多数のユーザに対してサービスを提供するような応用が増えている．しかし，従来のIDとパスワードによる認証などでは，未知のユーザに対する適切な権限割り当てが困難である．そこで，ユーザが持つ信任状に基づいて権限割り当てを行う技術である信用管理(trust management)[1,2,3,5]の必要性が生じてきた．信用管理とは，「A社の信任状を持っているユーザにはサービスRを提供してよい」のような信用管理ポリシーを予め定め，ユーザが提示したデジタル信任状（以下，単に信任状という）とポリシーを照合することで権限割り当てを行う手法のことである．このようにすることで，ユーザのIDやパスワード情報を予めシステムに登録することなく，適切な権限割り当てが可能となる．信任状は運転免許証，クレジットカード，学生IDなどの紙の信用証明書を電子化したものである．信任状は，それの所有者がある属性を持つことを保証する．例えば，学生は彼らが所属する大学から，その大学の学生であることを認める信任状を受け取ることができると考えられる．そして，彼らがオンライン書店の学生割引の資格を持つ学生であることを立証するのにその信任状を使用することができる．信任状は，第三者がその正当性を検証できるように電子的に署名されている．

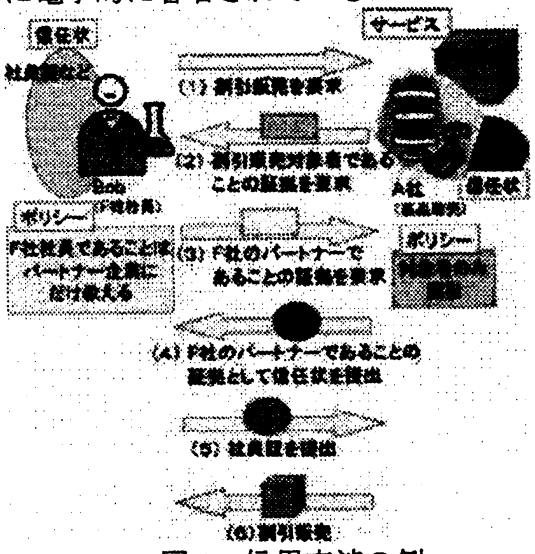


図1 信用交渉の例

信用管理のアプローチのひとつとして信用交渉(trust negotiation) [4,6-10]がある．信用交渉とは，サービス提供者とサービス要求者の両者が信任状の交換を繰り返して徐々に信用を確立する手法である．サービス提供者がユーザに一方的に信任状を要求する通

常の信用管理手法では，ユーザが予めサービス提供者を信用していることが前提となるが，信用交渉では全く知らない相手との間で信用を確立することができる．図1に信用交渉の例を示す．

図1では，交渉者はA社（薬品販売会社）とBob（F社社員）である．交渉者はそれぞれ信任状集合，ポリシー，サービスを持っている．ここでBobがA社に割引販売（A社のサービス）を求めたとする（図 1(1)）．するとA社はA社のポリシー「対象者のみ割引」に基づき，Bobに対して割引販売対象者であることの証拠を要求する（図1(2)）．Bobは割引販売対象者である証拠として自身の社員証を使うことができるが，Bobのポリシーで「F社社員であることはパートナー企業にだけ教える」となっている．そのため，まず先にA社にF社のパートナーであることの証拠を要求する（図 1(3)）．A社はその証拠として自身の信任状をBobに公開する（図1(4)）．Bobは自分のポリシーが満たされたので，A社に社員証を提出する（図1(5)）．このように信任状とポリシーの公開を繰り返し，可能な場合はBobに割引販売が与えられる（図1(6)）．不可能な場合は交渉が失敗に終わる．

信用交渉における戦略とは，それまでに交換した信任状と自身のポリシーを入力とし，次に相手に公開する信任状を返す写像である．交渉者は，計算量，交渉の長さ，公開する情報の量などの評価尺度のうち，交渉者自身が重視したい評価尺度に基づいて，戦略を選ぶことが出来る．Yuら[10]は，DTファミリーという戦略の集合を提案した．そして，二人の交渉者A, BがDTファミリーの中からそれぞれの戦略を選んで使用するならば，① AとBの交渉は必ず終了する，②両者のポリシーに照らして，信用を確立することが理論的に可能ならば，交渉によって必ず信用が確立される，といった望ましい性質が満たされることを示した．

YuらはDTファミリーの要素である具体的な戦略として，Simple Strategy，Relevant Strategy，Disclosure Tree Strategy（公開木戦略,以下DTS）の3つを提案している．Simple Strategyは公開できる信任状およびポリシーをすべて公開する戦略，Relevant Strategyは交渉に関係しているものだけをすべて公開する戦略である．DTSは，交渉を前進させるために必要となる信任状およびポリシーの極小集合を公開する戦略である．相手に公開する情報の量は，DTSがDTファミリーの中で最も少なく，DTS ≤ Relevant ≤ Simpleという関係になっている．すなわち，DTSはDTファミリーの中で最も慎重な戦略である．一方，DTSは単純な実装では指数的な計算量がかかるが，効率のよい実装が存在するかどうかや，計算量の下界は，これまで知られていなかった．

本研究では，Yuらの枠組の再定式化を行い，DTSの計算量の上界および下界について考察する．Yuらの定義では，DTSは極小集合である解をすべて出力し，ユーザがその中からひとつを選んで相手に公開するとなっているが，これを「解のうちひとつを出力する」としても実用上は支障ないと考えられる．そこで本研究では，解のうちひとつを出力する問題を扱う．この問題の計算量の下界が示せれば，すべての解を出力する問題の計算量はそれ以上であることが言える．

本論文ではまず，Yuらの定義に従うと，解のうちひとつを出力する場合でも，DTSはNP困難であることを示す．次に，以下のように問題の設定を変更することを考える．

(1) 冗長な公開を許す．例えば，以下のようなポリシーが公開済みであるとする．

「信任状A1は，B1,B2の両方またはB3が公開されたら公開してよい．」

「B1は，A1が公開されたら公開してよい．」

このとき，A1を公開可能にすることには，B2自身やB2のポリシーの公開は冗長である．Yuらの定義ではこのような冗長な公開は許さないようになっているが，この条件をなくす．

(2) 交渉を成功に導くことに貢献しない出力を行わない．すなわち，出力 m にさらに自分のポリシーの一部を加えると交渉が行き詰まるような出力は除外する．この場合，さらに別のポリシーの集合 m' を加えて公開することで交渉を再開できるが，初めから m' のみ公開するほうが適切である．

そして，(1)および(2)の一方だけではDTSのNP困難性は変わらないが，両方加えるとDTSは多項式時間可解であることを示す．

以降，2節で諸定義を行い，3節でDTSの計算量に関する結果を述べる．4節でまとめと今後の課題を述べる．なお証明の詳細については[11]を参照されたい．

2. 信用交渉と Disclosure Tree Strategy

ここでは，Yu ら[10]の信用交渉とDisclosure Tree Strategy（公開木戦略）の定義を整理し，[10]の定義を包含する形で再定式化を行う．

2.1. アクセス制御ポリシー

サービスや信任状を資源と呼ぶ．交渉者はそれぞれ有限個の資源を所有し，ひとつの資源を二者が共有することはないと仮定する．各資源はちょうど1つのアクセス制御ポリシー（以下，単にポリシーという）をもつ．ポリシーは「それが満たされたときのみ，その資源を相手に公開してもよい」という意味を表す．

ポリシーは以下のようにホーン節形式で記述する．

$$R \leftarrow (B_1 \wedge B_2) \vee B_3 \vee B_4$$

例： $B_1 \leftarrow A_2 \wedge A_3$

$$B_4 \leftarrow \mathit{false}$$

ポリシーの左辺はそのポリシーで保護されている資源，右辺はその資源を公開してもよい条件を表す論理式である．この論理式は，信任状と$\vee$，$\wedge$で構成される式か，空か，またはfalseである．例えば，上記のRを左辺とするポリシーは，信任状B_1とB_2が公開されたとき，またはB_3，またはB_4が公開されたとき，Rを公開してもよいという意味を表す．現在までに公開された信任状によってポリシーの右辺が満たされているとき，その資源はその時点でunlockedであるという．

ポリシーの右辺に現れる信任状の所有者は，左辺の資源の所有者とは必ず異なるとする．また，ポリシーの右辺は積和形とする．右辺が空の場合は，その資源はいつでも公開してよいという意味を表す．読みやすさのため，右辺が空のとき，$A\leftarrow$と書く代わりに，$A \leftarrow \mathit{true}$と書く．

右辺がfalseであるポリシーをdenialポリシーという．これは，その資源は決して公開しない（その資源を持っていない場合を含む）という意味を表す．denialポリシー以外のポリシーを許可ポリシーという．

2.2. 信用交渉

信用交渉では，交渉者はサービス提供者（ここではAliceとする）とサービス要求者（ここではBobとする）の二人である．BobがAliceにサービスRを要求することで交渉が始まる．

Aliceは，

i. Rを提供する．
ii. 自分の信任状の部分集合，および自分のポリシーの部分集合を相手に公開する．
iii. 交渉を打ち切る．

のいずれか一つを行う．iまたはiiで公開するサービスや信任状はunlockedでなければならない．

Aliceの選んだ動作がiiならば，次にBobはii，iiiのいずれかを行う．

以降，Rが提供されるか，交渉が打ち切られるまで交互に動作を行う．

1節での交渉例を使って説明する．交渉者はBobとA社である．それぞれ以下のようなサービス，信任状，ポリシーを持っている．

	Bob（F社社員）	A社（薬品販売）
サービス	—	R（割引販売）
信任状	B_1（社員証）	A_1（F社のパートナー企業であることを表す信任状）
ポリシー	$B_1 \leftarrow A_1 \vee A_2$	$R \leftarrow B_1 \vee (B_2 \wedge B_3)$，$A_1 \leftarrow \mathit{true}$，etc

交渉の流れは以下のとおりである．

(1) BobがA社に，Rを要求する（交渉開始）．
(2) A社はBobに，ポリシー$R \leftarrow B_1 \vee (B_2 \wedge B_3)$を公開する（$R$はポリシーで保護されているため，まずポリシーを公開する）．
(3) BobはA社に，$B_1 \leftarrow A_1 \vee A_2$を公開する（$B_1$はポリシーで保護されているため，まずポリシーを公開する）．
(4) A社はBobに，A_1を公開する（$A_1 \leftarrow \mathit{true}$なので公開可能）．

(5) Bob は A 社に，B_1 を公開する（B_1 のポリシーが満たされたので公開可能）．
(6) A 社は Bob に，R を提供する（R のポリシーが満たされたので提供可能，R が提供されたので交渉終了）．

戦略とは，それまでに公開された信任状集合と自身のポリシー全体の集合を入力として，次に相手に公開する信任状集合を返す写像である．交渉者双方の戦略に制約がなければ，交渉が停止するとは限らない．Yu ら[10]は DT ファミリーという戦略の集合を定義し，両交渉者が DT ファミリーの中の戦略を使用する（一方の交渉者が使用する戦略が他方のそれと異なっていてもよい）ならば，以下の性質が満たされることを示した．

- 交渉は必ず終了する．
- サービス R の提供が理論的に可能である（戦略に関係なく，R が unlocked となるような信任状とポリシーの公開操作列が少なくともひとつ存在する）ならば，交渉は R の提供で終わる．

DTS（2.5 節）は DT ファミリーの中で，相手に公開する情報の量が最小，すなわちもっとも慎重な戦略である．一方，DTS の効率的な実装（実行アルゴリズム）が存在するかどうかや計算量の下界はまだ知られていない．本論文では DTS の計算複雑さについて考察し，結果を 3 節で述べる．

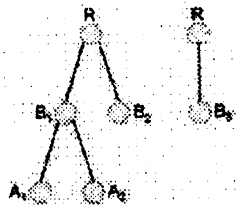
2.3. 公開木

両交渉者のポリシーの部分集合，および公開済み信任状の集合が与えられたとき，それらのポリシーによって作られる信任状間の依存関係を表した木構造を公開木（disclosure tree）という．例えば，図 2 は以下のポリシー集合に対する公開木の例である．

Aliceのポリシー
$R \leftarrow (B_1 \wedge B_2) \vee B_3$

Bobのポリシー
$B_1 \leftarrow A_1 \wedge A_2$

ポリシー集合の例



対応する公開木

図 2　公開木の例

許可ポリシーの集合 P，denial ポリシーの集合 P_d，および公開済み信任状の集合 C に対する公開木は，頂点に資源名をラベル付けした以下のような根付き木である．

- 根のラベルは R．
- u を任意の頂点，A を u のラベルとする．u と A は以下を満たす．
  - ➢ A は C に含まれない．
  - ➢ A のポリシーは P_d に含まれない．
  - ➢ A のポリシーが P に含まれ，ポリシーの右辺が空でないならば，右辺中のある積項 $B_1 \wedge \cdots \wedge B_k$ について以下が成り立つ．
    - ✧ $\{B_1,\ldots,B_k\}-C=\{B_{i_1},\ldots,B_{i_m}\}$ とする．このとき，u の子の集合は $\{u_1,\ldots,u_m\}$ かつ $1 \le j \le m$ について，u_j のラベルは B_{i_j}．つまり，u の子のラベルは，A のポリシーを満たす信任状のうちまだ公開されていないものである．特に $\{B_1,\ldots,B_k\} \subseteq C$ のときは，u は葉となる．
  - ➢ A のポリシーが P に含まれない，または A のポリシーの右辺が空ならば，u は葉である．
- ただし，有限の木のみ扱う．

P，P_d，C に対する公開木すべての集合を $view(P \cup P_d \cup C)$ と書く．資源 A が公開木の葉にラベル付けされているとき，A は A の所有者にとって evolvable であるという．また，その公開木，およびその公開木を含む集合も A の所有者にとって evolvable であるという．交渉者は，evolvable な信任状自身またはそのポリシーを次に公開することで，公開木を変形し，交渉を進展させることができる．「相手にとって evolvable な木が少なくともひとつ残る」ように自分が公開するものを決めるのが，2.5 節で述べる戦略 DTS の概略である．

木において，根から葉までのパス上に同じラベルが 2 回以上現れるとき，その木は冗長であるという．Yu らの DTS の定義では冗長でない公開木のみ扱われているが，3.2 節で述べるように，その場合 DTS は NP 困難である．3.2 節では，公開木の定義に「冗長でない」という条件を加えた場合，加えない場合，

それぞれについて DTS の計算複雑さを調べる.

2.4. 文脈自由文法による公開木集合の表現

許可ポリシーを文脈自由文法（CFG）の文法規則とみなすことで，公開木をその CFG における導出木とみなすことができる．すなわち，許可ポリシーの集合 P，denial ポリシーの集合 P_d，公開済み信任状の集合 C に対する CFG G を以下のように定義する．

- 資源 A のポリシーが P に含まれない，または A のポリシーの右辺が空ならば，A を終端記号とする．それ以外の資源を非終端記号とする．
- R を開始記号とする．
- G の文法規則の集合は，以下を満たす最小の集合．
  - ✧ P 中に資源 A のポリシー $A \leftarrow \cdots \vee (B_1 \wedge \cdots \wedge B_k) \vee \cdots$ が含まれ，かつ $\{B_1,\ldots,B_k\} - C = \{B_{i_1},\ldots,B_{i_m}\}$ ならば，$A \rightarrow B_{i_1} \ldots B_{i_m}$ が G の文法規則に含まれる（$\{B_1,\ldots,B_k\} \subseteq C$ のときは，規則の右辺は空文字列となる）．

このとき，$P \cup P_d \cup C$ に対する公開木集合は G の導出木集合と一致する．従って，$view(P \cup P_d \cup C)$ が空かどうかは，G が生成する言語 $L(G)$ が空かどうか調べることで判定できる．また，$view(P \cup P_d \cup C)$ が Alice にとって evolvable かどうか，すなわち Alice の信任状を葉とする公開木があるかどうかは，冗長な公開木を許すならば，Alice の信任状を含む終端記号列を G が導出できるかどうか調べることで判定できる．これは正規言語と $L(G)$ の共通集合が空かどうかを判定することで行える．ただし，冗長な公開木を許さない場合は，Alice の信任状を葉とする非冗長な導出木があるかどうか調べる必要がある．3.2 節で示すように，この問題は NP 完全である．

2.5. Disclosure Tree Strategy

Alice が Bob に次のメッセージを送ろうとしている時点を考える．メッセージとは，相手に公開（または提供）する資源およびポリシーの集合である．ただし，空メッセージは交渉打ち切りを表す．このとき，DTS は以下の入出力で定義される写像である．DTS の出力はメッセージの集合である．Alice は DTS の出力のうち任意のひとつを選んで相手に送信する．

入力：

- S_M　これまでに公開された信任状およびポリシーの集合
- L_A　Bob に公開していない Alice のポリシーの集合
- R　交渉の目的であるサービス

出力：

(1) ①『$view(S_M \cup L_A) = \varnothing$』，または②『$view(S_M)$ が Alice にとって evolvable でない』なら $\{\varnothing\}$ を返す．そうでないならば，

(2) ③『R が S_M の信任状で unlocked』なら $\{R\}$ を返す．そうでないならば，④『$view(S_M \cup m')$ が Bob にとって evolvable であるような，空でない m' すべて』を返す．ただし，m' の空でない真部分集合はこの条件を満たさない（すなわち m' は極小）．

Bob が Alice に次のメッセージを送ろうとしているときも同様である．

なお，本研究では扱わないが，DT ファミリーの他の戦略として，DTS 以外に Simple Strategy と Relevant Strategy がある．Simple Strategy は，未公開の自分のポリシーと unlocked である信任状をすべて相手に公開する．すべてのポリシーおよびすべての unlocked である信任状を既に公開していた場合，すなわち新たに公開できるものが何もない場合は $\{\varnothing\}$ を返す．Relevant Strategy は，evolvable であるすべての信任状について，unlocked ならばそれ自身，そうでなければそのポリシーを相手に公開する．

3. DTS の計算複雑さ

3.1. 問題の定義

DTS は以下の 4 ステップに分解できる．

① $view(S_M \cup L_A) = \varnothing$ かどうかを判定．

② $view(S_M)$ が Alice にとって evolvable かどうかを判定．

③ RがS_Mの信任状でunlockedかどうかを判定.

④ $view(S_M \cup m')$がBobにとってevolvableであるような極小なm'すべてを求める.

このうち③は,Rのポリシーと公開済み信任状から容易に行える.また2.4節で述べたように,①は文脈自由言語の空判定によって行える.そこで,以降では②と④のみ扱う.以降,②をEVL,④をMsetと書くことにする.さらに,④において「すべてのm'を答える」とした場合をA-Mset,「条件を満たすm'のうちひとつを答える」とした場合をS-Msetとする.

公開木の定義に冗長でないという条件を加えた場合は,問題名の前にNR-を付けて表す.加えない場合は,問題名の前にR-を付けて表す.なお,NR-問題で答えが「evolvable」または「m'が存在する」であった場合はR-問題でも同じ答えになるが,そうでないときはR-問題について何も言えない.従って,「R-問題よりNR-問題のほうが難しい(R-問題はNR-問題に帰着可能)」とは言えない.

3.2. DTSのNP困難性

定理1: NR-EVLはNP完全である.

証明: NP可解性の証明は省略する.NP困難性を3SATからの帰着によって示す.3SATのインスタンスを

変数集合 $\{x_1, x_2, \ldots, x_m\}$

節集合 $\{c_1, c_2, \ldots, c_n\}$

とする.上記インスタンスに対し,以下のようなポリシーからなる集合Pを考える.

$$R \leftarrow B_{x_1}$$

$$\left.\begin{array}{l} B_{x_i} \leftarrow A_{x_i} \vee A_{\overline{x_i}} \\ A_{x_i} \leftarrow B_{x_{i+1}} \vee B_0 \\ A_{\overline{x_i}} \leftarrow B_{x_{i+1}} \vee B_0 \end{array}\right\} \quad 1 \le i \le m$$

$$B_{x_{m+1}} \leftarrow A_1$$

$$A_1 \leftarrow B_{c_1} \wedge B_{c_2} \wedge \ldots \wedge B_{c_n}$$

$$\left.\begin{array}{ll} B_{c_j} \leftarrow A_0 \wedge A_{\overline{x_i}} & \text{if } x_i \in c_j \\ B_{c_j} \leftarrow A_0 \wedge A_{x_i} & \text{if } \overline{x_i} \in c_j \end{array}\right\} \quad 1 \le j \le n$$

上記より,公開木の葉になり得るAliceの信任状はA_0のみである.そして,$view(P)$にA_0を含む非冗長公開木が存在するとき,かつそのときのみ,上記3SATのインスタンスは充足可能である. □

定理2: R-EVLは$O(n)$時間可解である.ただし,nは$S_M \cup L_A$の記述長である.

証明: 2.4節の議論より,文脈自由言語と正規言語の共通集合が空かどうか判定することに帰着できる.この正規言語は状態数が定数個の有限オートマトンで表せるので,題意が言える. □

定理3: NR-S-MsetおよびR-S-MsetはNP困難である.

証明: 3SATからの帰着によって示すことができる. □

各問題の計算複雑さをまとめると以下のようになる.

	NR	R
EVL	NP完全	$O(n)$
S-Mset	NP困難	NP困難

ただし,S-MsetのNP困難性は「$view(S_M \cup L_A)$が空でない」という条件を外して(つまり,「②を通過したときのみ④を実行する」という条件を無視して)証明している.「$view(S_M \cup L_A)$が空でない」という条件を加えたときにS-Msetの計算複雑さがどうなるかは不明である.

3.3. 妥当な制約の下での多項式時間可解性

公開木の定義,およびDTSが返すm'の定義を,以下の(a),(b)のように変更することを考える.

(a)公開木の定義に以下の条件を加える.

公開済み信任状によって信任状Aのポリシーが満たされているとき,Aがラベル付けされた頂点は葉(子を持たない)とする

例えば,Aのポリシーが$A \leftarrow (B_1 \wedge B_2) \vee (B_3 \wedge B_4)$で,$B_1$と$B_2$が公開済みの

とき，B_3, B_4 を A の子とするような木は許さないことにする．すでに A は公開可能であるのにさらにほかの信任状を要求するのは無意味なので，このように定義するのは合理的と考えられる．このように定義すると，unlocked な信任状 A は必ず葉になるので，公開木の葉以外の頂点が信任状の公開によって消されることはない．

(b) m' の条件「$view(S_M \cup m')$ が Bob にとって evolvable」を以下のように強める（以下の条件を満たす m' のうち極小のものを解とする）．

「$m' \subseteq m'' \subseteq m' \cup L_A$ である任意の m'' について，$view(S_M \cup m'')$ が Bob にとって evolvable」

つまり，自分の未公開ポリシーをさらに公開したとき，相手にとって evolvable でなくなるような m' は解としない，という意味である．例えば，以下の S_M と L_A を考える．

$S_M = \{R \leftarrow (B_1 \wedge B_2) \vee B_3,\ B_1 \leftarrow A_1,\ B_2 \leftarrow A_2,\ B_3 \leftarrow A_3\}$
$L_A = \{A_1 \leftarrow B_4,\ A_2 \leftarrow B_2,\ A_3 \leftarrow B_3\}$

このとき，$m' = \{A_1 \leftarrow B_4\}$ は Yu らの定義では解となるが，$m'' = \{A_1 \leftarrow B_4, A_2 \leftarrow B_2\}$ のとき $view(S_M \cup m'')$ が Bob にとって evolvable でないので，(b) の下では m' は解とならない．

(b) は，① で判定している条件 $view(S_M \cup L_A) \neq \varnothing$ をさらに強めたもの，と言うこともできる．「自分の未公開ポリシーを加えると相手にとって evolvable でなくなる」ということは，そのような m' は R を公開可能にするのに役に立たないということなので，そのような m' を排除するのは合理的と考えられる．

定理 4： 条件 (a), (b) の下で，R-S-Mset は $O(n^2)$ 時間可解である．ただし，n は $S_M \cup L_A$ の記述長である．

証明： 資源を頂点とする有向グラフを考え，ポリシー未公開の Alice の信任状 x（または $x=R$）について，x のポリシーの右辺から公開済みポリシーによって到達できる資源 y に対して辺 (x,y) を置く．このグラフを到達性グラフと呼ぶ．Bob の資源のうち，ポリシーの右辺の信任状すべてが unlocked なものの集合を V_R とする．到達性グラフにおける R から V_R の要素までの極小なパス（パス上で隣り合っていない頂点間には辺がない）が，R-S-Mset の解と一致することを示せる．このようなパスは深さ優先探索によって見つけることができる．到達性グラフの構築，および極小なパスの発見はいずれも $O(n^2)$ 時間で行える． □

条件 (a), (b) を仮定した場合，各問題の計算複雑さは以下のようになる．

	NR	R
EVL	NP 完全	$O(n)$
S-Mset	NP 困難	$O(n^2)$

4. まとめ

本研究では，Yu らが提案した信用交渉戦略である DTS に関して，Yu らの枠組の再定式化を行い，DTS の計算量の上界および下界について考察した．具体的には，Yu らの定義に従うと DTS は NP 困難であること，また，交渉を成功に導くことに貢献しない出力を除外するよう条件を変えた場合には多項式時間可解であることを示した．

今後の課題として，各種評価尺度（計算量，交渉の長さ，公開する情報の量など）の下で，既知の DT ファミリーの戦略（Simple Strategy, Relevant Strategy, DTS）のうち，どの戦略が優れているのか調査することが挙げられる．各戦略の特性を明らかにすることにより，交渉者は自身が重視する尺度に従って戦略を選ぶことができる．さらに，望ましい特性を持つ新しい戦略を開発するための基礎となる．

文 献

[1] M. Blaze, J. Feigenbaum and J. Lacy, "Decentralized Trust Management," IEEE Security and Privacy, pp.164--173, Jun 1996.

[2] M. Blaze, J. Feigenbaum, J. Ioannidis and A. Keromytis, "The KeyNote Trust-Management System," RFC 2704, Sep 1999.

[3] A. Herzberg, Y. Mass, J. Mihaeli, D. Naor and Y. Ravid, "Access Control Meets Public Key Infrastructure, or, Assigning Roles to Strangers," IEEE Security and Privacy, pp.2--14, May 2000.

[4] H. Koshutanski and F. Massacci,

"Interactive Credential Negotiation for Stateful Business Processes," the 3rd International Conference on Trust Management, pp.256--272, May 2005.

[5] S. Ruohomaa and L. Kutvonen, "Trust Management Survey," 3rd International Conference on Trust Management, pp.77--92, May 2005.

[6] W. H. Winsborough and N. Li, "Towards Practical Automated Trust Negotiation," the 3rd IEEE International Workshop on Policies for Distributed Systems and Networks, pp.92--103, June 2002.

[7] W. H. Winsborough and N. Li, "Safety in Automated Trust Negotiation," IEEE Security and Privacy, pp.147--160, May 2004.

[8] T. Ryutov, L. Zhou, B. C. Neuman, T. Leithead, K. E. Seamons, "Adaptive Trust Negotiation and Access Control," SACMAT 2005, pp.139--146, Jun 2005.

[9] M. Winslett, T. Yu, K.E. Seamons, A. Hess, J. Jacobson, R. Jarvis, B. Smith, L. Yu, "Negotiating Trust on the Web," IEEE Internet Computing, Vol.6, No.6, pp.30--37, Nov/Dec 2002.

[10] T. Yu, M. Winslett and K. E. Seamons, "Supporting Structured Credentials and Sensitive Policies through Interoperable Strategies for Automated Trust Negotiation," ACM Transactions on Information and System Security, Vol.6, No.1, pp.1--42, Feb 2003.

[11] 山本有輝, "信用交渉における公開木戦略の計算量," 修士論文 NAIST-IS-MT0551132, 奈良先端科学技術大学院大学情報科学研究科, Feb 2007.